

REMARKS

The above amendment and these remarks are responsive to the Final Office action of 26 Nov 2004 by Examiner Peter J. Smith.

Claims 1, 3-5, 7-8, 10-12, and 14-20 are in the case, none as yet allowed.

Applicant's invention relates to data passing in an environment which is described her specification as follows:

In accordance with a preferred embodiment of the invention, in a Notes/DB2 hybrid environment, a Notes agent reads data from a DB2 table, and then dynamically populates that data to an HTML page. In this manner, an the existing infrastructure (including Notes, Domino Go, and DB2) is used to combine HTML and Java script web presentation with DB2 data.

[Specification, page 54.]

For the purpose of the description of this

embodiment, RCW exists in a frame set in a browser. A frame set divides a screen into logical and user-friendly sections called frames. As part of security for browser 100, Netscape and Internet Explorer (IE) establish ownership of the frames: each frame is a window under control of browser 100. Browser 100 can open up a session in any frame desired. However, if all of the frames open on a browser are not owned by same session, then these frames cannot see the contents of each other.

Consequently, the problem is presented: if RCW needs to access an outside supplier site 300 for information to get back through one its own the frames, as soon as data is written by that other site into one of the frame at browser 100, RCW no longer owns the frame and cannot access the information. So this aspect of the invention is concerned with data passing.

In accordance with the preferred embodiment of the invention, the supplier 300 opens a new window through normal Java script commands and writes its data into it, along with the name of the agent to run in RCW application server 114. That new window then calls one

of the RCW processes, which is able to see the data because it is not in a frame and is therefore available for RCW to access. That process causes an agent to runs (in RCW) that can see that data and write it to a frame on the RCW side, resulting in RCW owning both the data and the frame. This allows RCW to reach NOTES information, and write that information to other frames. RCW owns the data and the frame. Ownership is established in the RCW application.

Consequently, by carefully tracking and controlling the sequence of opening windows, RCW can establish the origin from which data is coming, where the data is going, and which process owns it.

Those are the high points. Review: two sites are talking to each other using an intermediate window, doing a handoff of data and of ownership of both data and frame such that when it is time to supply information to the RCW application, RCW owns the data, frame and process.

Referring to Figure 21, a system and method are provided for data transfer from a externally owned site

to an application owned frame set which operates as follows. The RCW application opens a supplier site URL in a temporary frame 456. Upon user selection of go to supplier 451, as is represented by step 458, frame 456 opens the supplier window 460 as a separate browser session. Two windows are now open: the original application 450 with its window underlying, and a supplier window 460 over it with the supplier URL. The reason for doing this is that the supplier requires that the browser be full frame, not in a small frame set. The primary RCW application in window 450 is quiesced to a wait state. As is represented by steps 462, the user can now select from window 460 items to buy, search, or whatever the supplier deems is appropriate for a user to order his data. The user then issues the command to submit the order. In step 444, the supplier site then gathers content from order data entered at window 460, in step 446 formats the page, and in step 448 issues a call to Req/Cat Web to open third window 464 with first agent 480. The supplier uses an enterprise specified agent name for first agent 480 when opening third window 464. Third window 464 is a window, but not a frame, and thus the Req/Cat Web can get access to it even though it is

opened by the supplier. First agent 480 includes an html form command 488 which defines the processing to be done on the contents 466 of the form now displayed in third window 464, and the supplier site writes into this third window 464 unique order identifying information. Once written, the browser activates the form. Once activated, it is a program in its own right, the html 488 that was written and any java script in it will execute. One of first things it does is look at the action in the form command and determine that this is the program that will run to deal with the contents of this form. That action program, or first agent 480, is a RCW action program on the RCW server that can see contents of third window 464 because it is not in a frame, and thus ownership is not critical. In step 482, first RCW agent 480 executes a program or process that writes the contents 466 of window 464 back into its frame set (temporary frame) 456, and then calls second agent 484 which references Java script code 454 and, as is represented by step 486, access Notes data on the Notes server, add content to the requisition, and issue the commands that send the order to be stored in the requisition. This is key, Req/Cat Web has used its own process 480 to write into its own

window 450, and knows who owns the data. First agent 480 process opened window 464 and writes the data to temporary frame 456 and then kicks off another process, second agent 484, that can read that data, can read and write to all of the frames 452 in the application, can access information from Lotus notes, and write all the information into the requisition.

[Specification, page 60,
line 8 forward.]

35 U.S.C. 103

7. Claims 1, 3, 15, and 16-17 have been rejected under 35 U.S.C. 103(a) over Lloyd, US 6,460,041 B2 in view of Hills et al. (hereinafter, Hills), U.S. Patent 6,239,797 B1 and Kanavy et al. (hereinafter, Kanavy), U.S. Patent 5,544,298.

As the Examiner states, "Lloyd does not teach executing a temporary frame in a first browser window" [Office action, page 3], and introduces "...the prior art reference of Hills et al. to teach the added claim limitation of a temporary data frame used to dynamically populate data to a page

display." and "...the prior art reference of Kanavy et al. to teach the added claim limitation of opening a new window which is used to operate an agent to collect data from the database." [Office action, page 8.]

Applicant has amended claims 1, 3, 15, and 16-17 to set forth the sequence of opening windows in order to establish the origin from which data is coming, where the data is going, and which process owns it in an environment where frame spoofing is inhibited.

Lloyd relates to a browser-based database-access apparatus and method. As pointed out by the Examiner, Lloyd teaches executing an agent to read data from a database table at Col. 2, line 56 ff.

Hills describes remote database record scroll/update without refresh. The Examiner refers to Hills at Columns 1 and 4, which teach:

The control application further instructs the client browser to maintain a hidden scroll frame such that data received from the database is placed into the scroll frame and then populated into the display frame

without refreshing the entire display frame that is visible to the user. The application further instructs the client browser to maintain a hidden update frame such that data intended to update the database is populated from the display frame into the update frame and then sent to the database without refreshing the entire display frame.

...In one embodiment, the application sends the data as an HTML page with accompanying JAVASCRIPT instructions.... an entire new HTML page does not have to be sent to the browser for loading the display frame. Only the necessary data to change the data contents of the existing display frame as well as instructions need to be sent.... the user's browser populates the data from the scroll frame into the display frame without refreshing the entire screen.

There is in Hills no description of the specific windows, frames, and agents set forth in applicant's claims. Applicant's claims require that the third window not be a frame, and that it include data from the second session and an agent from the first session, which are used to populate the data from the second session to a temporary data frame

owned by the first session, and that yet another agent refresh the HTML page with data from the temporary data frame. This is not taught by Hills.

Kanavy describes a code generation and data access system. The Examiner refers to Kanavy at Col. 4, which states:

A window 60 is created temporarily, listing database filenames 61. One or more of the database file names is selected by the user, either by keyboard entries or by mouse manipulations. At this point, one of the salient aspects of the system according to the invention is reflected in what is seen on the screen. As each database file 61 is selected, the application makes reference to the field name information for the selected file...

However, there is no suggestion in Kanavy of applicant's claimed third window not being a frame, and including data from a second session and an agent owned by a first session in an environment where frame spoofing is inhibited to pass data owned by a second session to populate a page display markup language owned by a first session.

The above mentioned distinctions of each individual reference, of course, do not address their combination. Applicant argues that the only manner in which these references may be combined to teach applicant's claims, as amended, would require hindsight reasoning using applicant's claims as a road map to assemble disparate teachings. Those teachings, of the three cited references, do not, for example, set forth in any reasonable combination the three windows, the two agents, the inhibition of frame spoofing, a third window which is not a frame and includes second session data and a first session agent, and so forth.

Consequently, applicant urges that the rejection of claims a, 3, 15, and 16-17 be withdrawn.

8. Claims 8 and 10-11 have been rejected under 35 U.S.C. 103(a) over Lloyd in view of Hills, Kanavy, and Lin et al. (hereinafter, Lin), U.S. Patent 6,052,785.

Applicant has amended independent claim 8 substantially as set forth above, such that claim 8 and its dependent claims 10 and 11 are distinguished as previously set forth

from Lloyd, Hills, and Kanavy, individually and as combined.

Lin relates to a sourcing system, but like Lloyd, Hills, and Kanavy, does not teach applicant's combination of windows, frames, and agents for accessing the database, as now claimed.

Applicant urges that claims 8 and 10-11 be allowed.

9. Claims 4 and 18 have been rejected under 35 U.S.C. 103(a) over Lloyd in view of Hills, Kanavy, and further in view of Lin.

Claim 4 depends from claim 1 and claim 18 depends from claim 15, and are distinguished from the combination of Lloyd, Hills, Kanavy and Lin as previously explained.

Applicant urges that claims 4 and 18 be allowed.

10. Claims 5, 7, 12, 14, and 19-20 have been rejected under 35 U.S.C. 103(a) over Lloyd in view of Hills, Kanavy, and Lin, and further in view of Johnson et al (hereinafter, Johnson) U.S. Patent 6,023,683.

These claims all depend from base claims distinguished from the combination of Lloyd, Hills, Kanavy and Lin as explained above.

Johnson relates to an electronic requisition catalog application including a relational database, and is cited by the Examiner as teaching "an electronic requisition catalog application implemented via a database table comprising a relational database for storing catalog data...." [Office Action, page 7.] However, Johnson does not teach applicant's specific combination of windows, frames, and agents for accessing that table. As that specific combination is not taught, as is explained above, by the combination of Lloyd, Hills, Kanavy and Lin, neither is it taught when Johnson is added to the combination.

Applicant requests that claims 5, 7, 12, 14, and 19-20 be allowed.

SUMMARY AND CONCLUSION


Applicant urges that the above amendments be entered and the case passed to issue with claims 1, 3-5, 7-8, 10-12, and 14-20.

The Application is believed to be in condition for allowance and such action by the Examiner is urged. Should differences remain, however, which do not place one/more of the remaining claims in condition for allowance, the Examiner is requested to phone the undersigned at the number provided below for the purpose of providing constructive assistance and suggestions in accordance with M.P.E.P. Sections 707.02(j) and 707.03 in order that allowable claims can be presented, thereby placing the Application in condition for allowance without further proceedings being necessary.

Sincerely,

A. J. Snavely

By


Shelley M Beckstrand
Reg. No. 24,886

Date: 26 Jan 2005

Shelley M Beckstrand, P.C.
Attorney at Law
61 Glenmont Road
Woodlawn, VA 24381-1341

Phone: (276) 238-1972